OIPE
MAR 1 4 2005
PATENT & TRADEMARK OFFICE

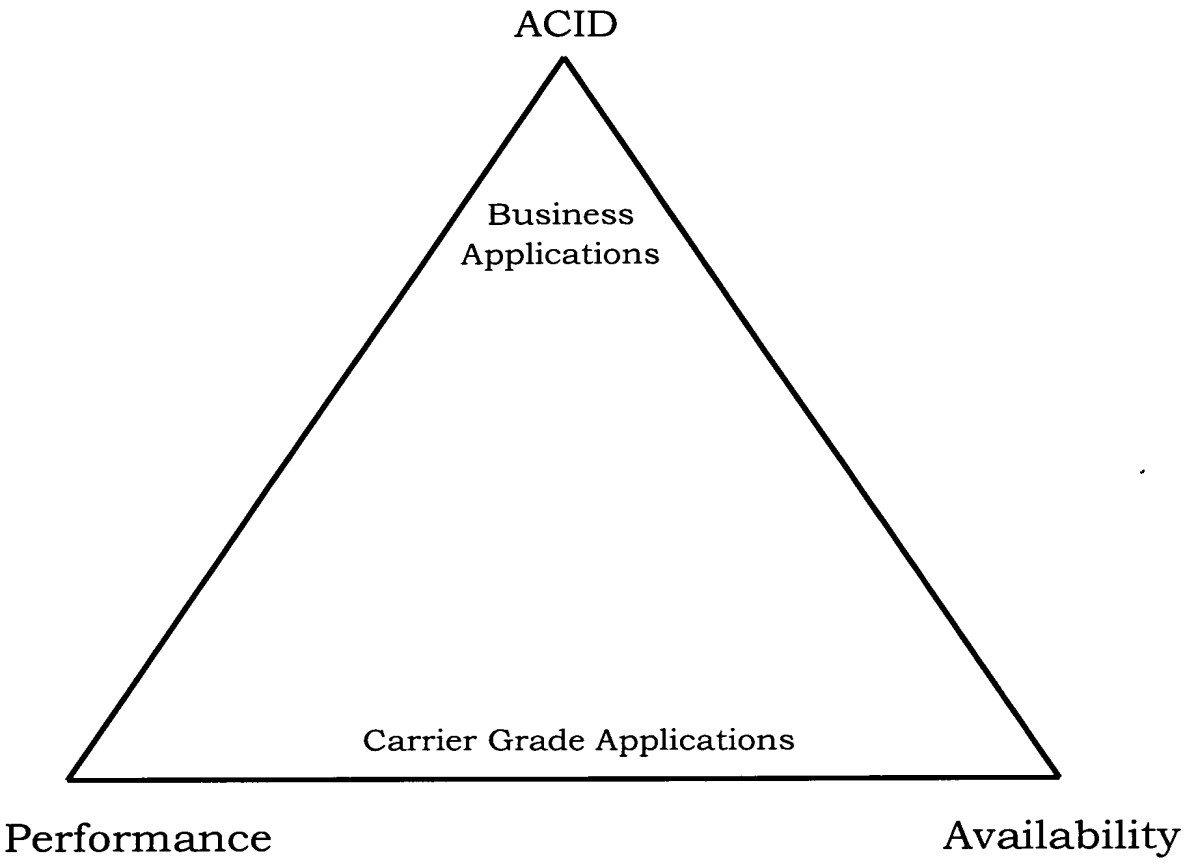

FIG. 1
(Prior Art)

OIPE
MAR 1 4 2005
PATENT & TRADEMARK OFFICE

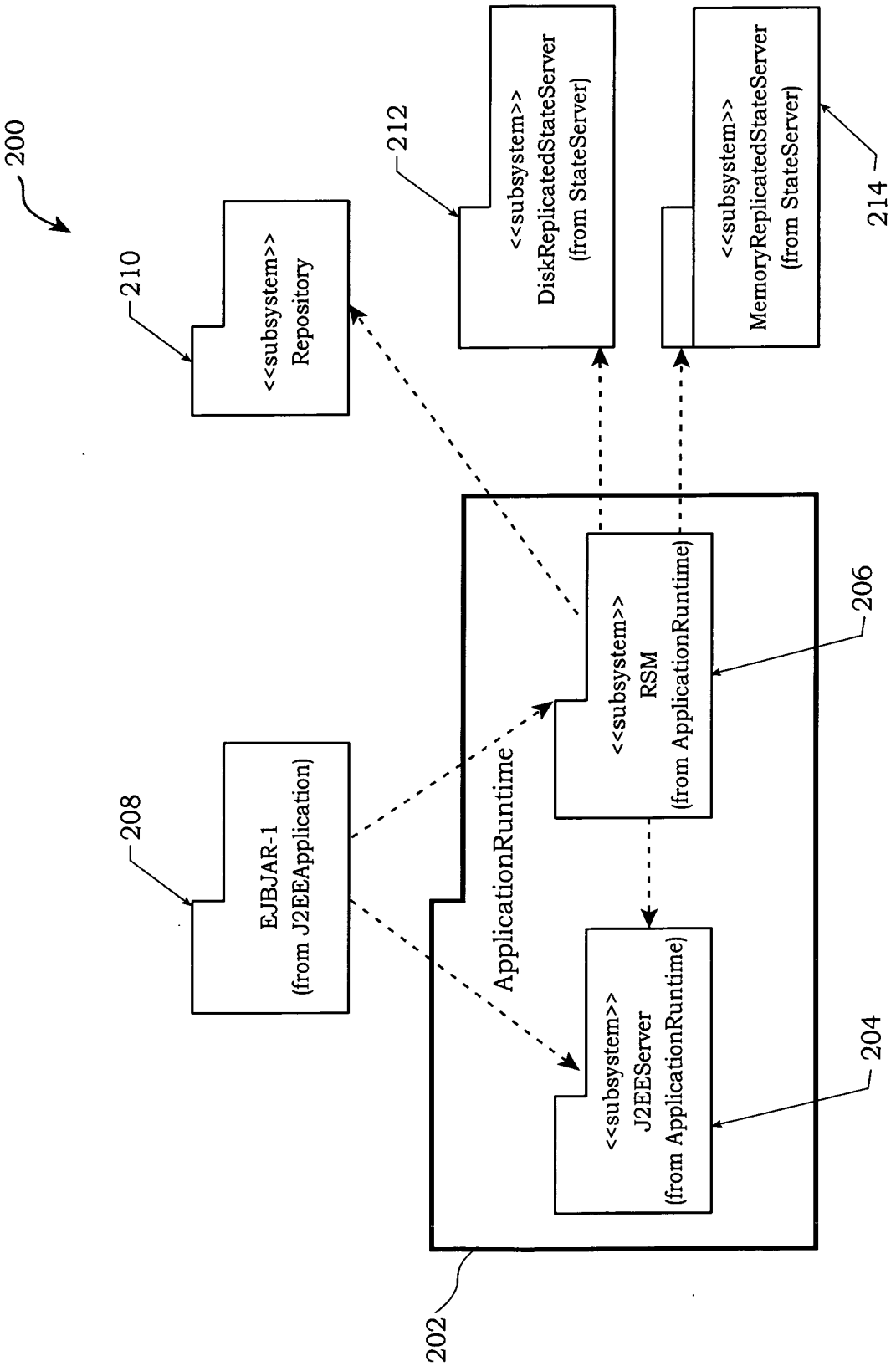

FIG. 2

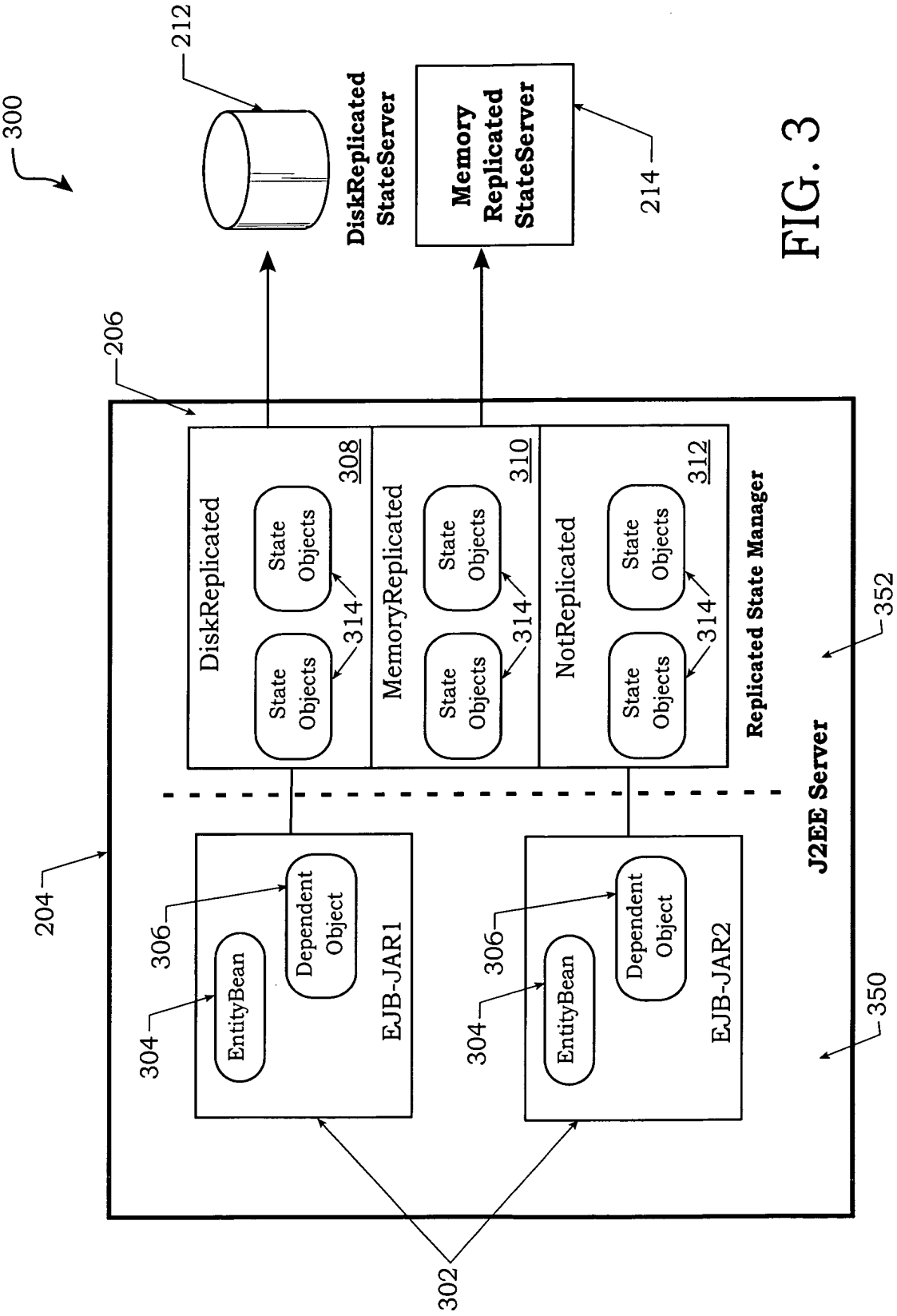
300
212
DiskReplicated StateServer
Memory Replicated StateServer
214
206
DiskReplicated
State Objects
308
314
MemoryReplicated
State Objects
310
314
NotReplicated
State Objects
312
314
Replicated State Manager
352
204
306
Dependent Object
304
EntityBean
EJB-JAR1
306
Dependent Object
304
EntityBean
EJB-JAR2
J2EE Server
350
302

FIG. 3

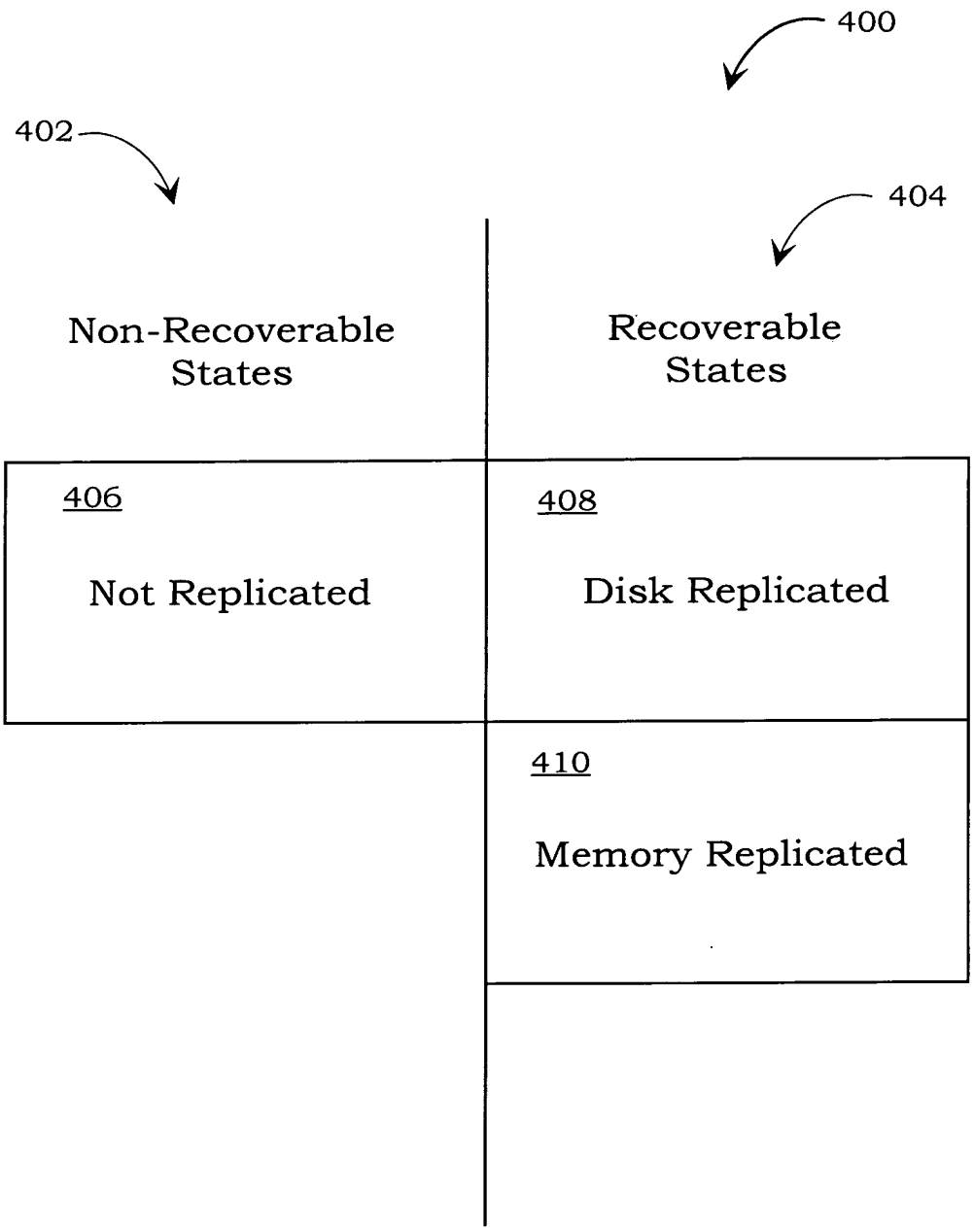
400
402
404
Non-Recoverable States
Recoverable States
406
Not Replicated
408
Disk Replicated
410
Memory Replicated

FIG. 4

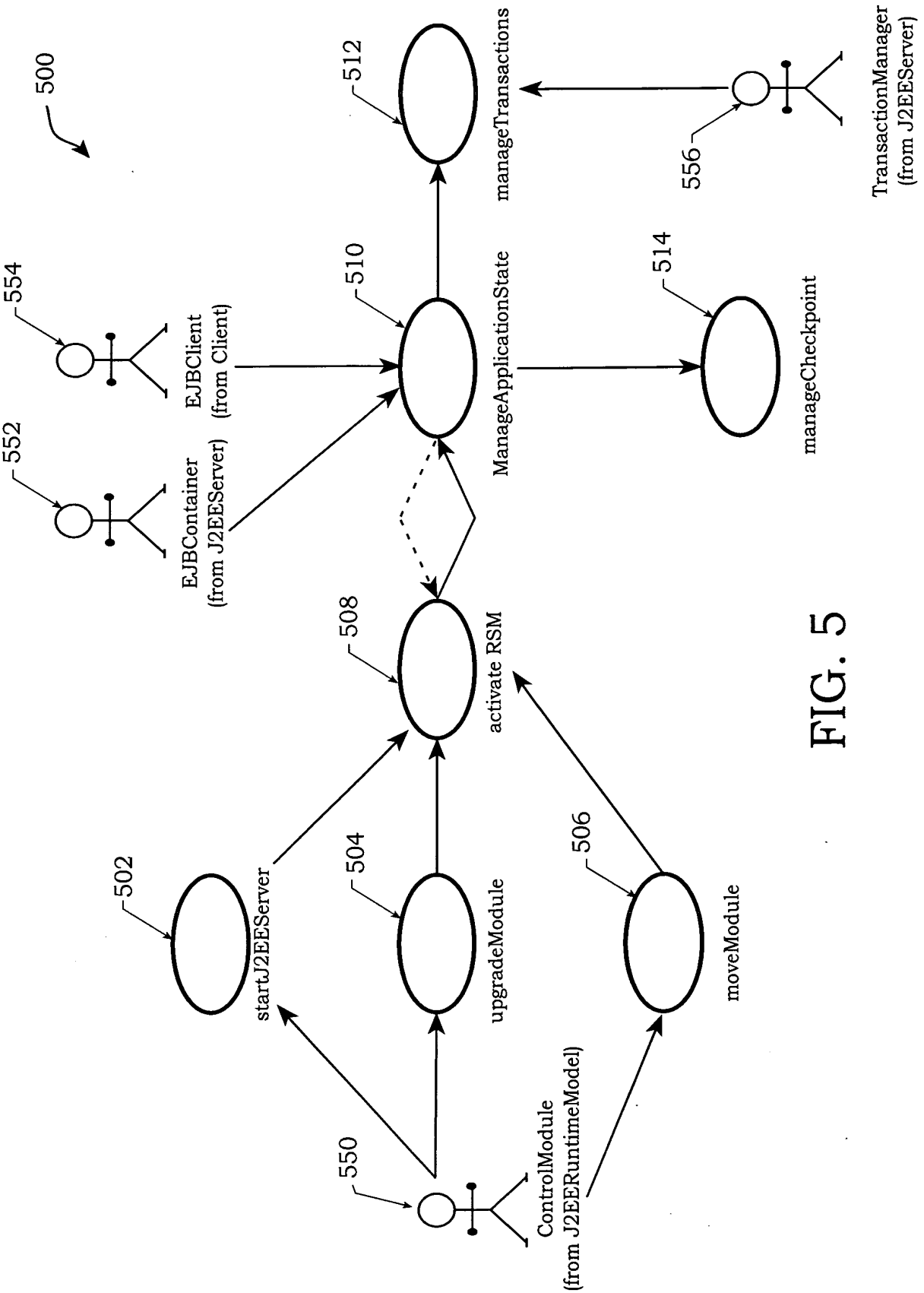
500
512
manageTransactions
556
TransactionManager
(from J2EEServer)
554
EJBClient
(from Client)
510
ManageApplicationState
514
manageCheckpoint
552
EJBContainer
(from J2EEServer)
508
activate RSM
502
startJ2EEServer
504
upgradeModule
506
moveModule
550
ControlModule
(from J2ERuntimeModel)

FIG. 5

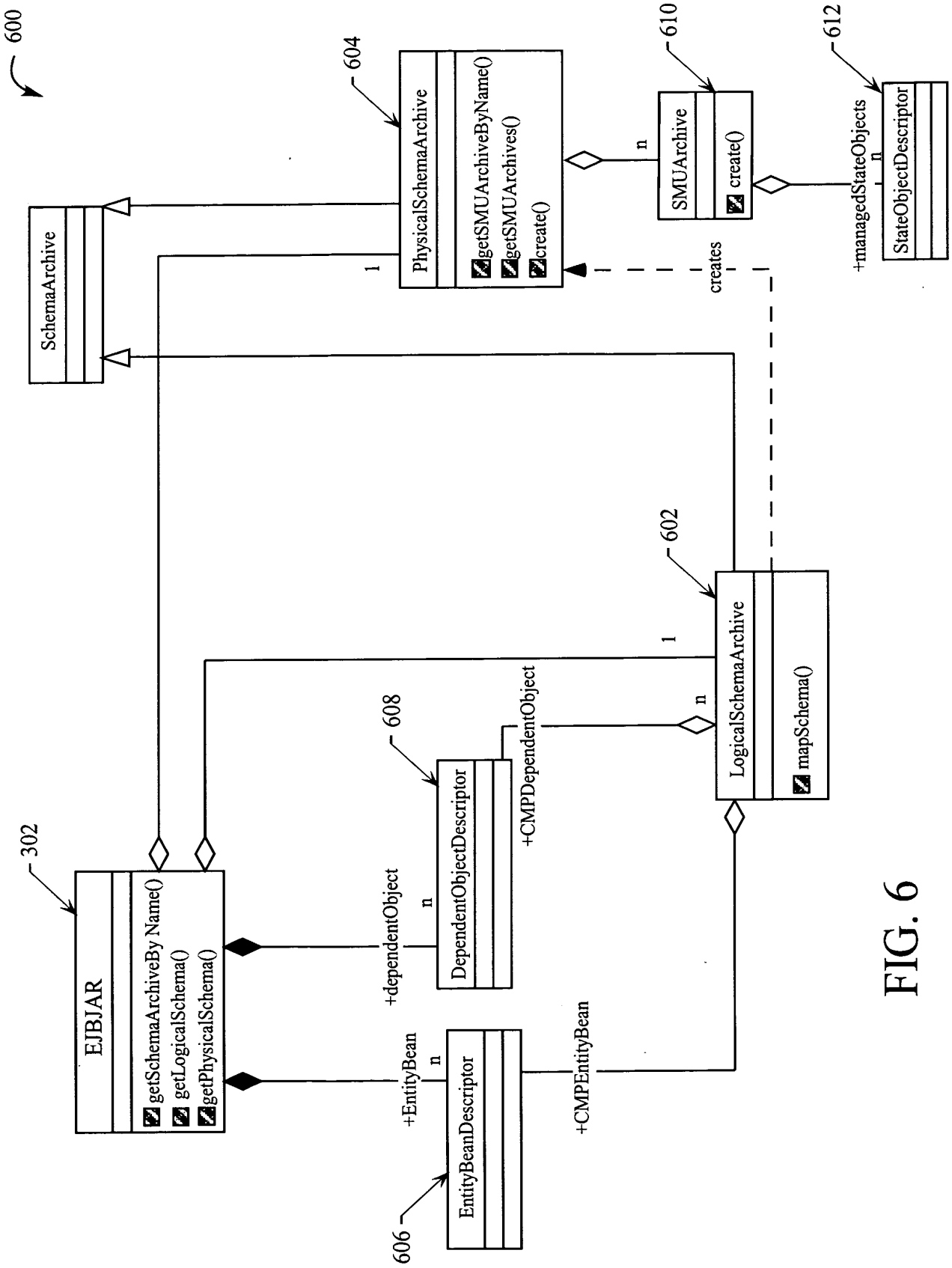
600
604
PhysicalSchemaArchive
getSMUArchiveByName()
getSMUArchives()
create()
610
SMUArchive
create()
612
StateObjectDescriptor
+managedStateObjects
SchemaArchive
creates
602
LogicalSchemaArchive
mapSchema()
608
DependentObjectDescriptor
+CMPDependentObject
+dependentObject
302
EJBJAR
getSchemaArchiveBy Name()
getLogicalSchema()
getPhysicalSchema()
+EntityBean
+CMPEntityBean
EntityBeanDescriptor
606

FIG. 6

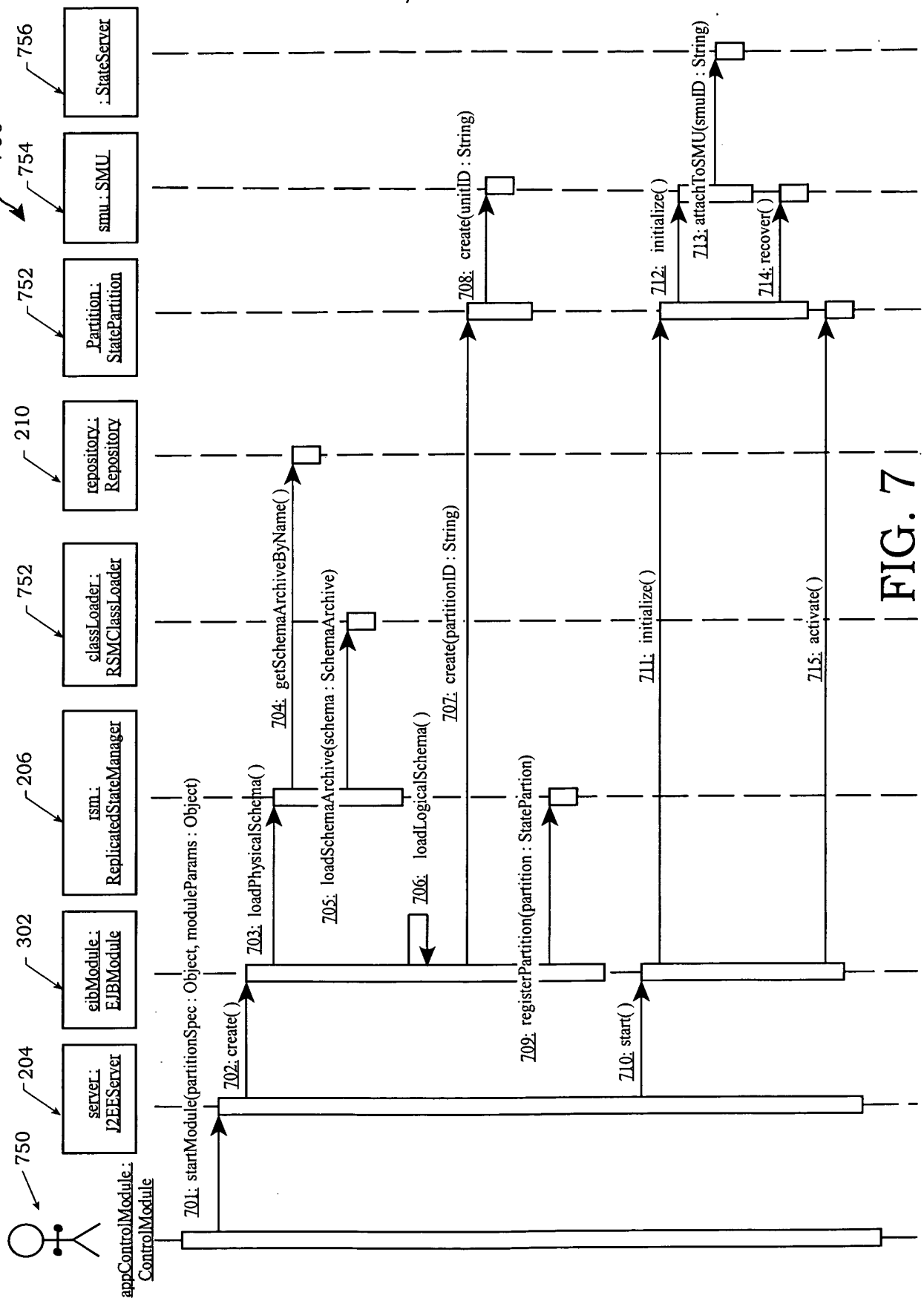
700
750
appControlModule : ControlModule
server : J2EEServer
204
ejbModule : EJBModule
302
rsm : ReplicatedStateManager
206
classLoader : RSMClassLoader
752
repository : Repository
210
Partition : StatePartition
752
smu : SMU
754
: StateServer
756
701: startModule(partitionSpec : Object, moduleParams : Object)
702: create( )
703: loadPhysicalSchema( )
704: getSchemaArchiveByName( )
705: loadSchemaArchive(schema : SchemaArchive)
706: loadLogicalSchema( )
707: create(partitionID : String)
708: create(unitID : String)
709: registerPartition(partition : StatePartion)
710: start( )
711: initialize( )
712: initialize( )
713: attachToSMU(smuID : String)
714: recover( )
715: activate( )

FIG. 7

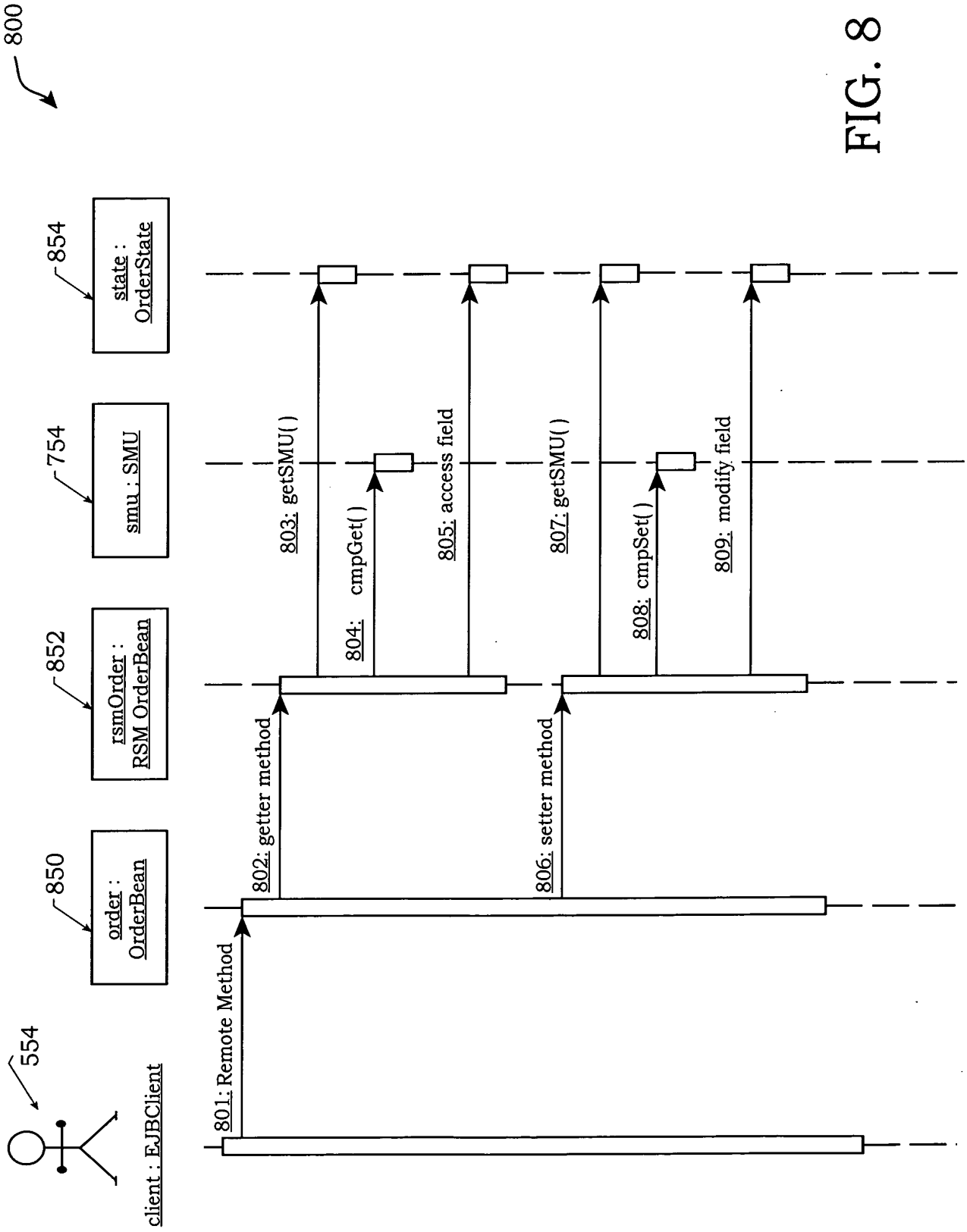
800
554
client : EJBClient
850
order : OrderBean
852
rsmOrder : RSM OrderBean
754
smu : SMU
854
state : OrderState
801: Remote Method
802: getter method
803: getSMU( )
804: cmpGet( )
805: access field
806: setter method
807: getSMU( )
808: cmpSet( )
809: modify field

FIG. 8

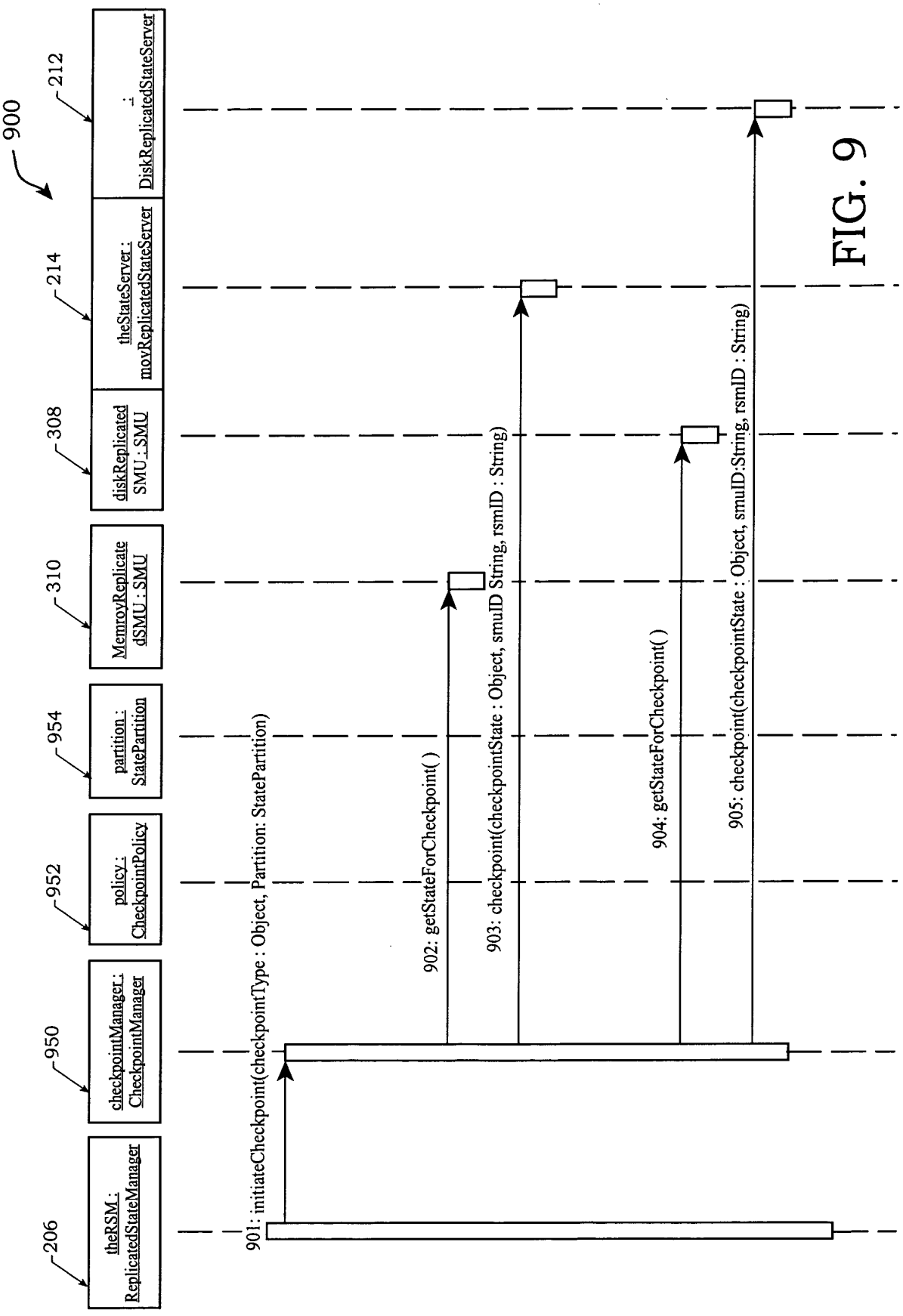
900
206
theRSM : ReplicatedStateManager
950
checkpointManager : CheckpointManager
952
policy : CheckpointPolicy
954
partition : StatePartition
310
MemroyReplicate dSMU : SMU
308
diskReplicated SMU : SMU
214
theStateServer : movReplicatedStateServer
212
: DiskReplicatedStateServer
901: initiateCheckpoint(checkpointType : Object, Partition: StatePartition)
902: getStateForCheckpoint( )
903: checkpoint(checkpointState : Object, smuID String, rsmID : String)
904: getStateForCheckpoint( )
905: checkpoint(checkpointState : Object, smuID:String, rsmID : String)

FIG. 9

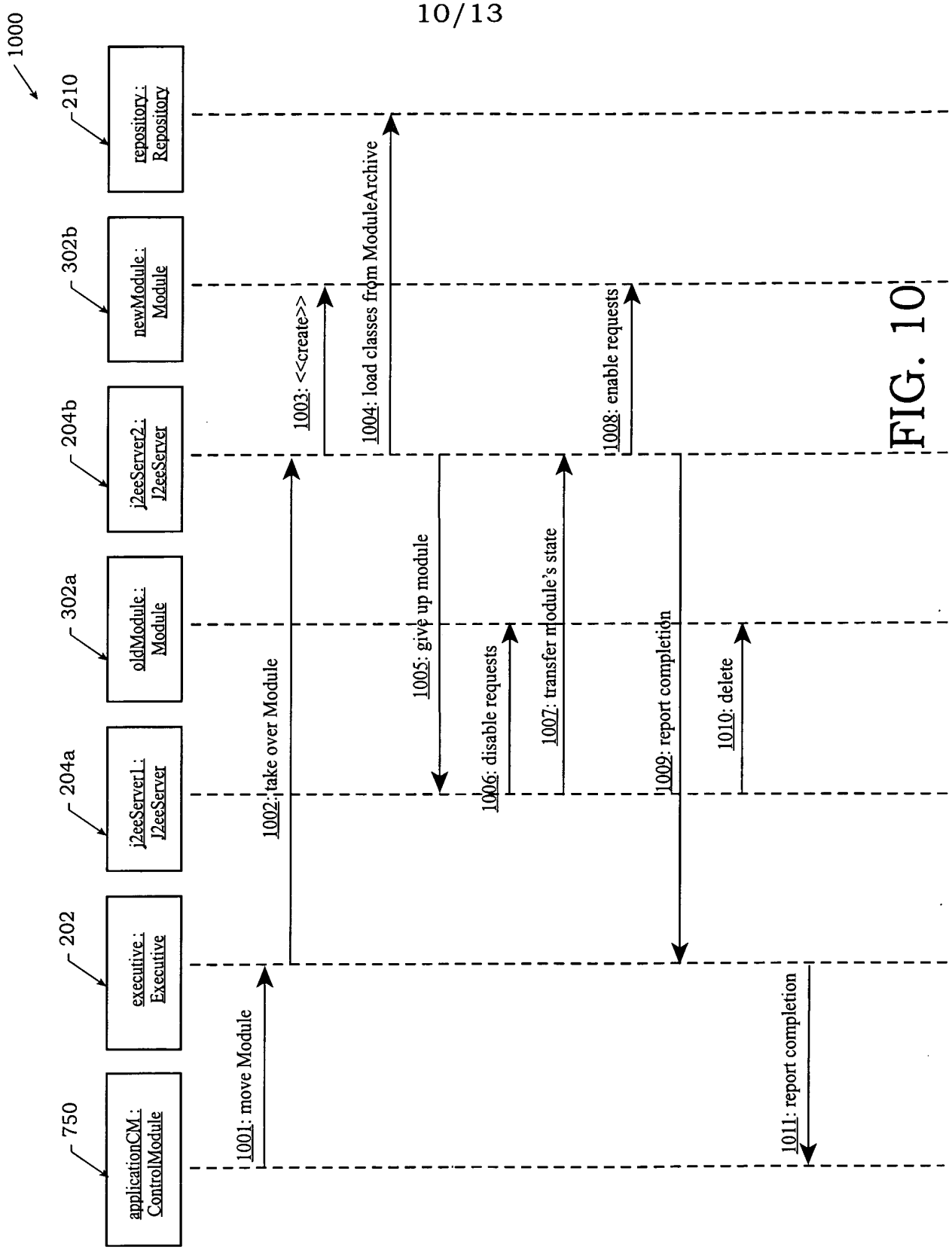
1000
750
applicationCM : ControlModule
202
executive : Executive
204a
j2eeServer1 : J2eeServer
302a
oldModule : Module
204b
j2eeServer2 : J2eeServer
302b
newModule : Module
210
repository : Repository
1001: move Module
1002: take over Module
1003: <<create>>
1004: load classes from ModuleArchive
1005: give up module
1006: disable requests
1007: transfer module's state
1008: enable requests
1009: report completion
1010: delete
1011: report completion

FIG. 10

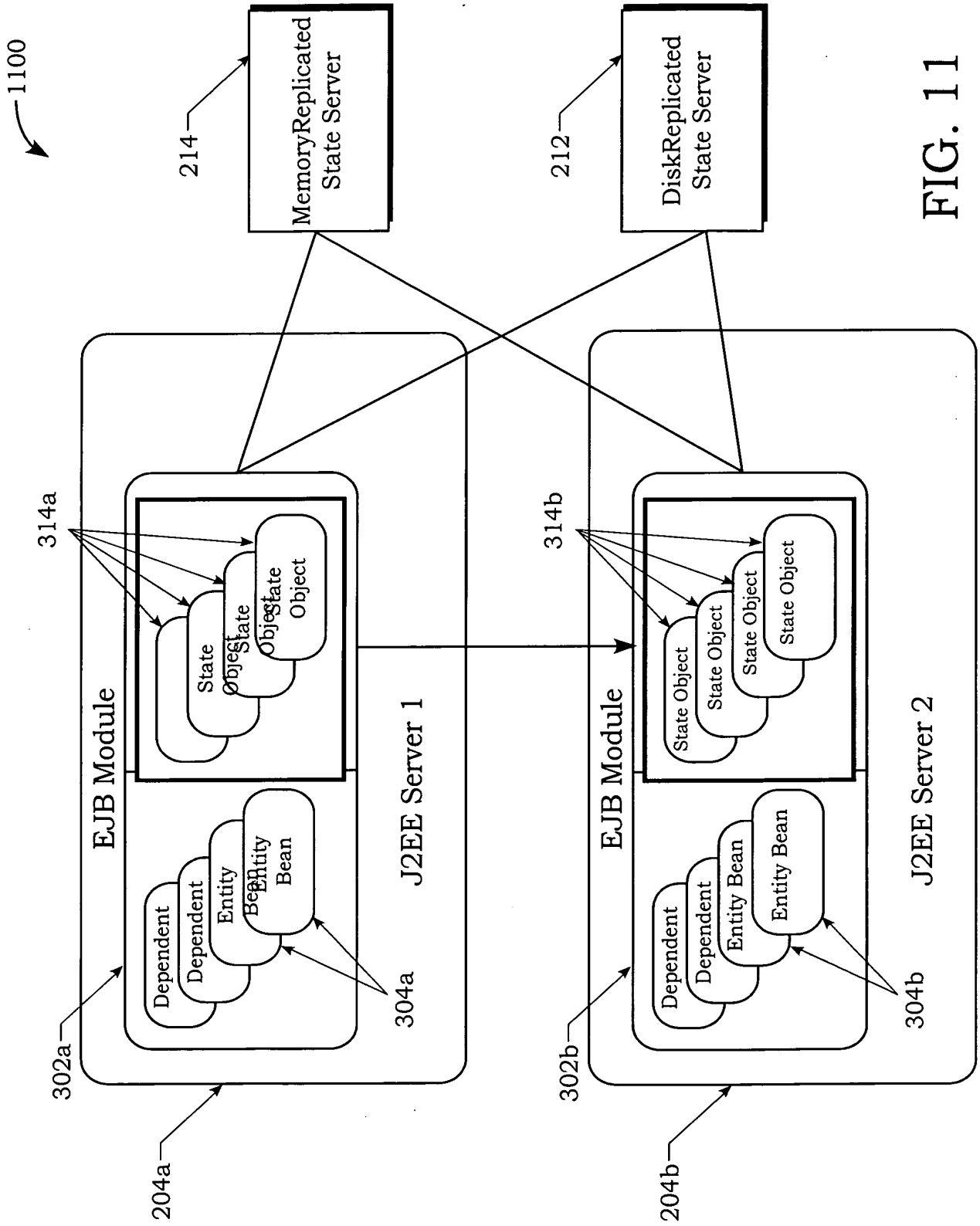
1100
214
MemoryReplicated State Server
212
DiskReplicated State Server
314a
314b
EJB Module
J2EE Server 1
J2EE Server 2
State Object
Dependent
Entity Bean
302a
302b
304a
304b
204a
204b

FIG. 11

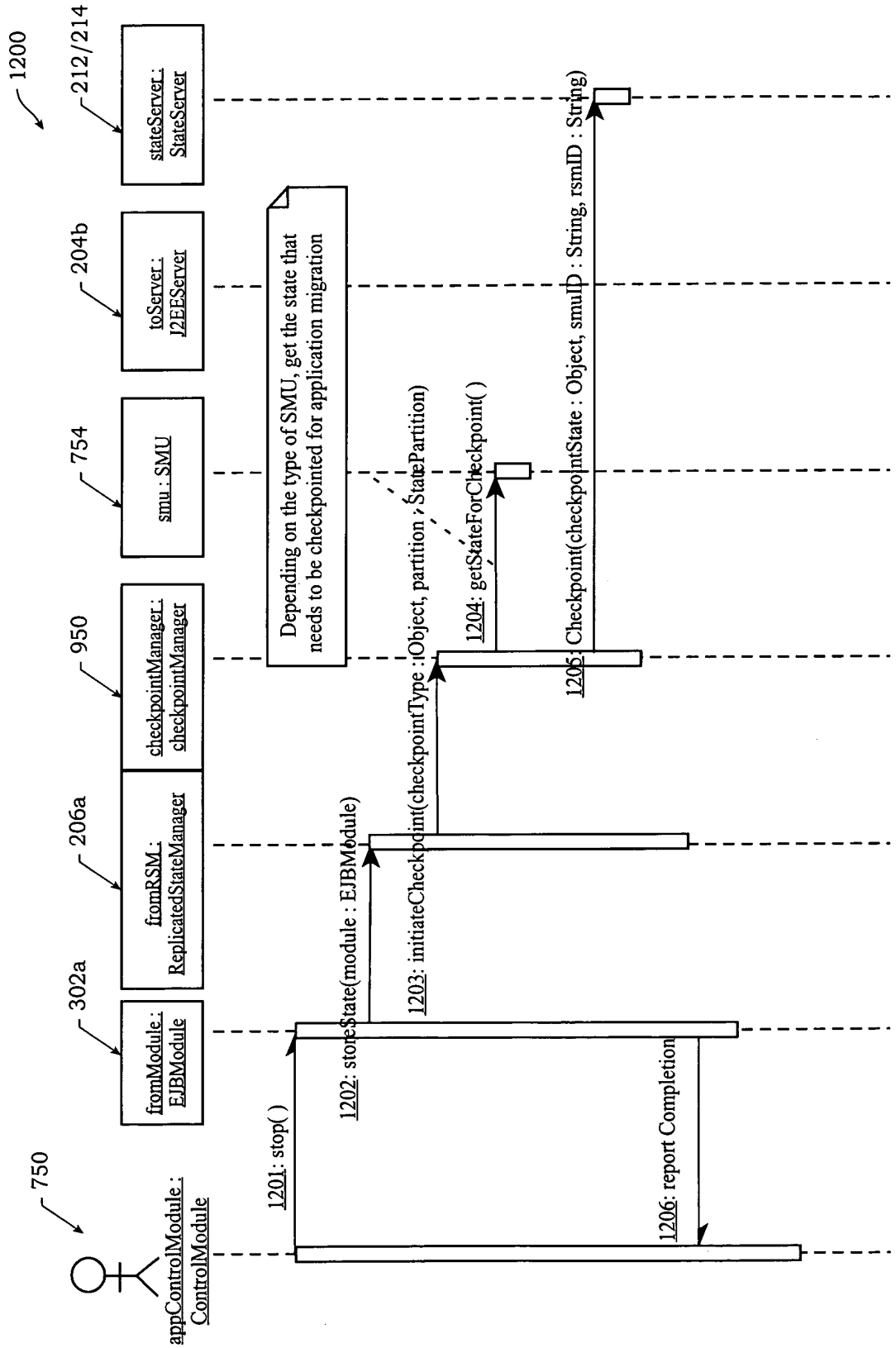
1200
750
appControlModule : ControlModule
302a
fromModule : EJBModule
206a
fromRSM : ReplicatedStateManager
950
checkpointManager : checkpointManager
754
smu : SMU
204b
toServer : J2EEServer
212/214
stateServer : StateServer
Depending on the type of SMU, get the state that needs to be checkpointed for application migration
1201: stop( )
1202: storeState(module : EJBModule)
1203: initiateCheckpoint(checkpointType :Object, partition : StatePartition)
1204: getStateForCheckpoint( )
1205: Checkpoint(checkpointState : Object, smuID : String, rsmID : String)
1206: report Completion

FIG. 12

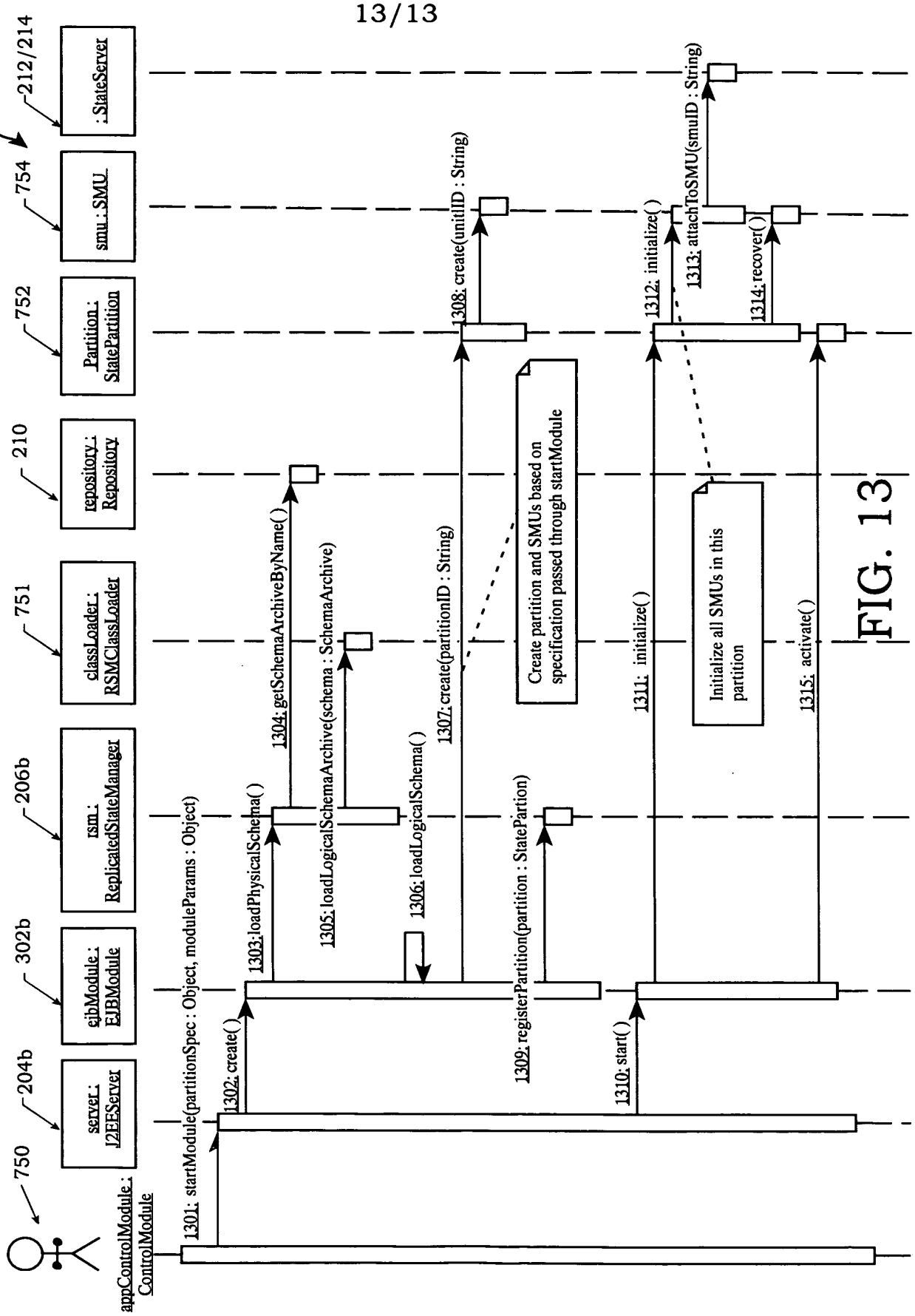
1300
750
appControlModule : ControlModule
204b
server : J2EEServer
302b
ejbModule : EJBModule
206b
rsm : ReplicatedStateManager
751
classLoader : RSMClassLoader
210
repository : Repository
752
Partition : StatePartition
754
smu : SMU
212/214
: StateServer
1301: startModule(partitionSpec : Object, moduleParams : Object)
1302: create( )
1303: loadPhysicalSchema( )
1304: getSchemaArchiveByName( )
1305: loadLogicalSchemaArchive(schema : SchemaArchive)
1306: loadLogicalSchema( )
1307: create(partitionID : String)
1308: create(unitID : String)
1309: registerPartition(partition : StatePartion)
Create partition and SMUs based on specification passed through startModule
1310: start( )
1311: initialize( )
1312: initialize( )
1313: attachToSMU(smuID : String)
Initialize all SMUs in this partition
1314: recover( )
1315: activate( )

FIG. 13